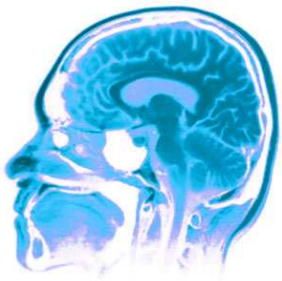
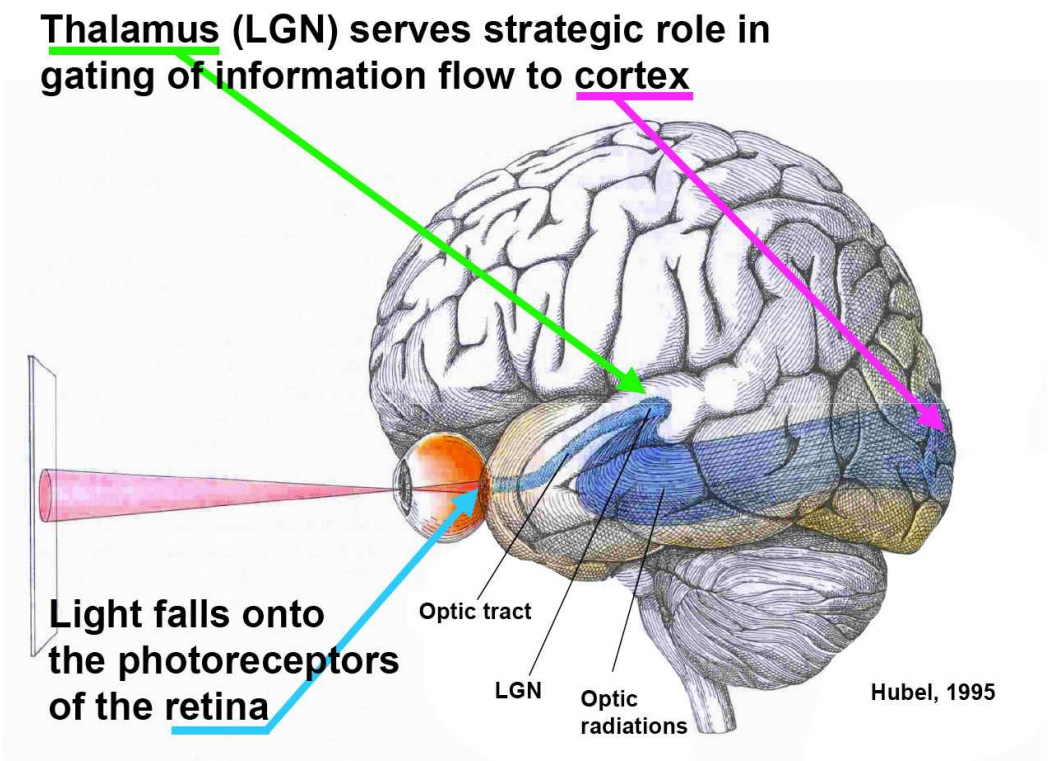
CPSC540

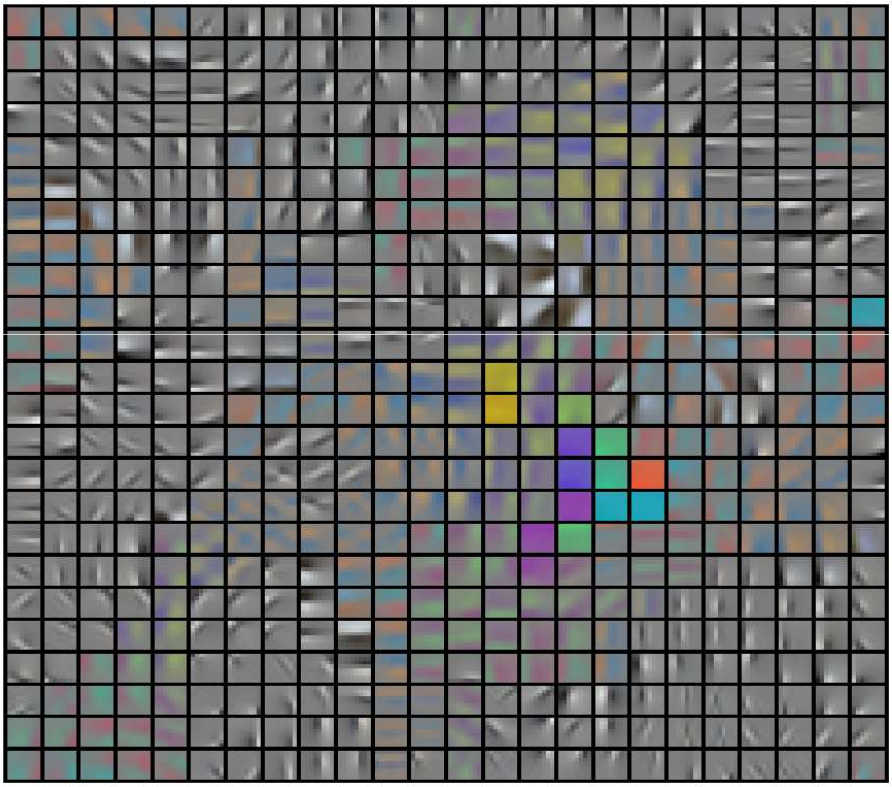


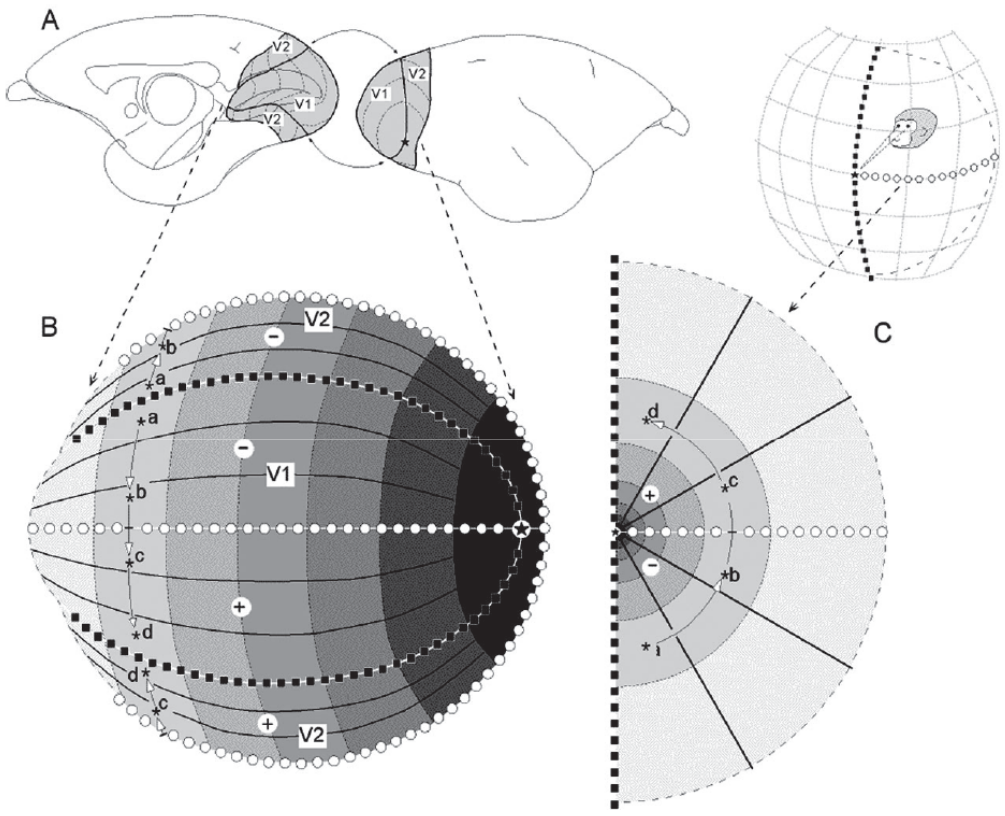
Boltzmann Machines and Satisfiability
Information, Computation & Energy

Nando de Freitas
October 2011

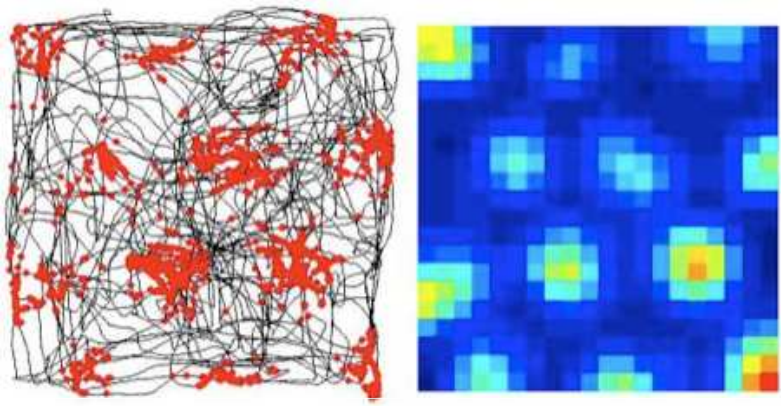


Topographic maps



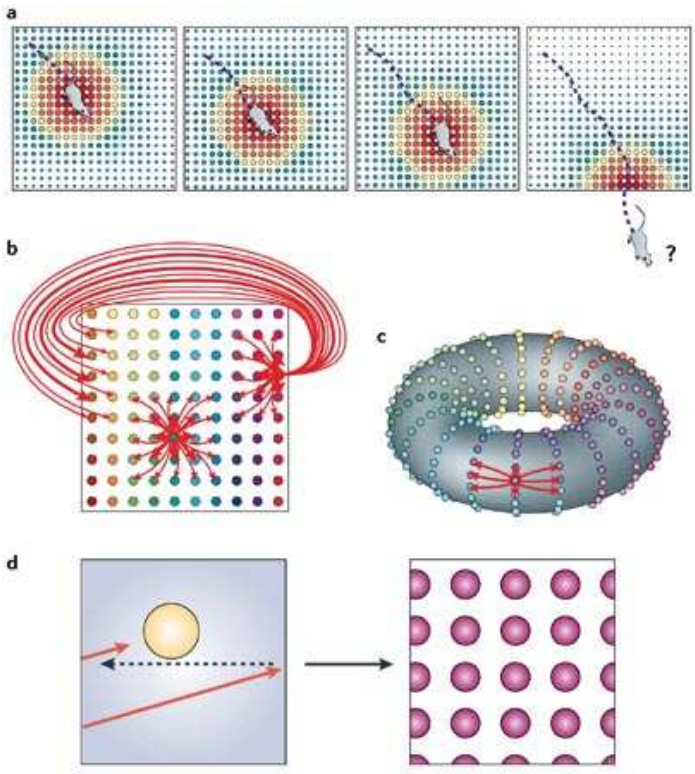


[Rosa, 2002]

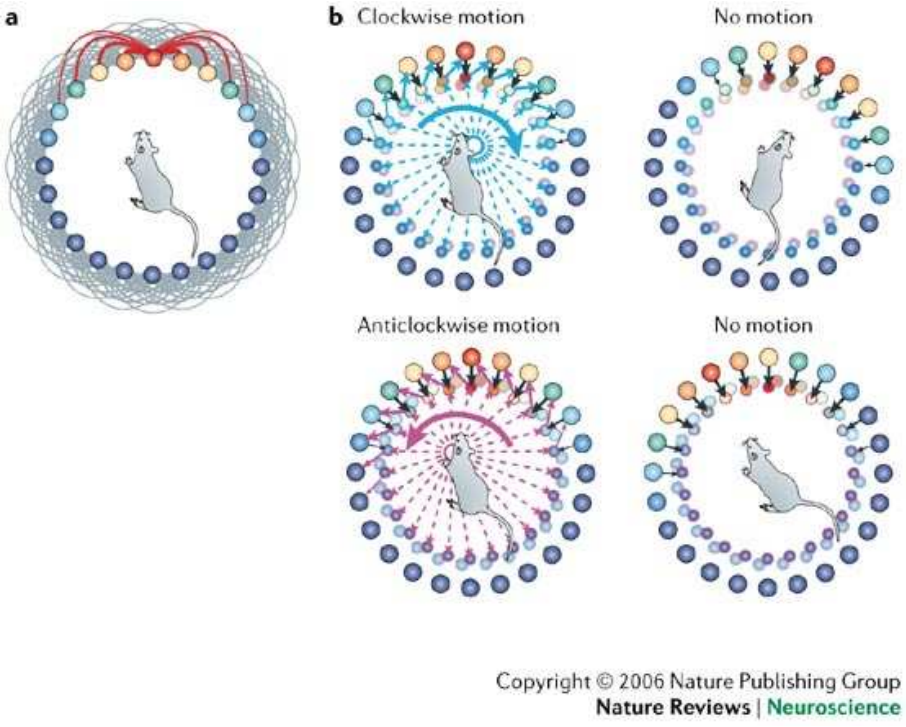


"*the x and y coordinates correspond to the spatial location of a rat, which is running around freely inside a large box. The black lines in the left figure shows how this particular rat explored the box in a fairly haphazard manner. However, an electrode inserted in the rat's subcortex picks up a signal that is anything but chaotic: the responses of said neuron are given as red dots in the left figure, while the right figure gives the firing rate distribution (ranging from blue for silent and red for the peak rate of responding). Although the rat is running about randomly, this neuron is responding in a grid, seemingly coming on an off in response to the animal's spatial location.*"

[Hafting et al 2005]

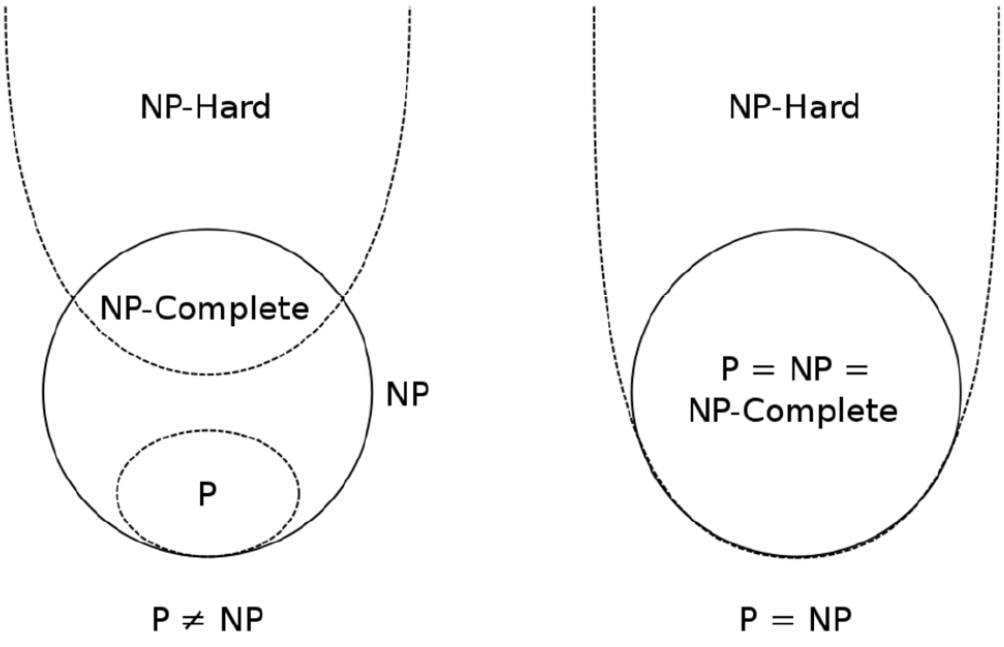


McNaughton *et al. Nature Reviews Neuroscience* 7, 663–678 (August 2006)



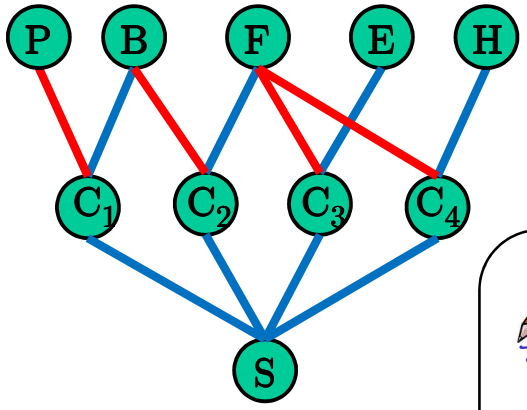
McNaughton *et al. Nature Reviews Neuroscience* 7, 663–678 (August 2006)

Computational Complexity



Re-visiting logic, NP and 2-SAT

Consider the CNF expression $S = C_1 \wedge \ldots \wedge C_m$, where each clause C_i is a disjunction of literals $x_{i,1} \vee \ldots \vee x_{i,k_i}$ defined on propositional variables. When each clause has two parents at most, the problem is known as 2-SAT.

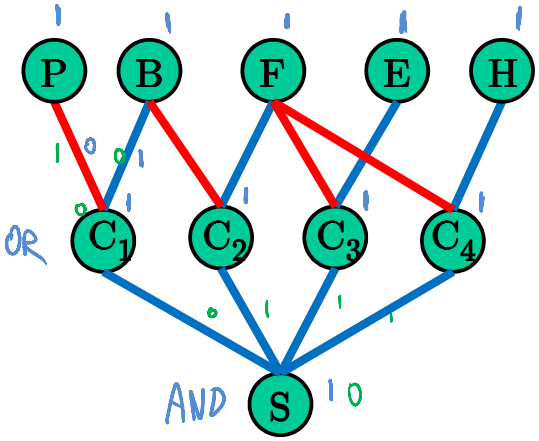


implication symbol

$$Parrot \implies Bird \qquad \neg P \vee B$$
$$Bird \implies Flies \qquad \neg B \vee F$$
$$Flies \implies Escapes \qquad \neg F \vee E$$
$$Flies \implies HasWings \qquad \neg F \vee H$$

P	B	$\neg P \vee B$ (imply)
0	0	1
0	1	1
1	0	0
1	1	1

Re-visiting logic, NP and 2-SAT $S = C_1 \wedge C_2 \wedge C_3 \wedge C_4$



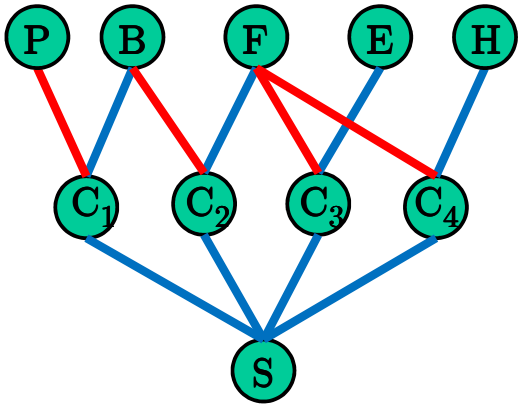
$$Parrot \implies Bird \qquad C_1 = \neg P \vee B$$
$$Bird \implies Flies \qquad C_2 = \neg B \vee F$$
$$Flies \implies Escapes \qquad C_3 = \neg F \vee E$$
$$Flies \implies HasWings \qquad C_4 = \neg F \vee H$$

$S = C_1 \wedge C_2 \wedge C_3 \wedge C_4$

P	B	F	E	H	C_1	C_2	C_3	C_4	S

1. **Verification: Does (P=1,B=1,F=1,E=1,H=1), i.e. (11111), satisfy this 2-SAT problem?** S = 1 YES
2. **Verification: Does (10111) satisfy it?** NO
3. **Maximization: What is the maximum number of clauses that can be satisfied?**
4. **What is the number of possible assignments to (PBFEH)?** 2^5
5. **Counting: How many assignments satisfy this 2-SAT example?**

Logic, NP, 2-SAT and Monte Carlo



#S=1 is 4

trials = 33

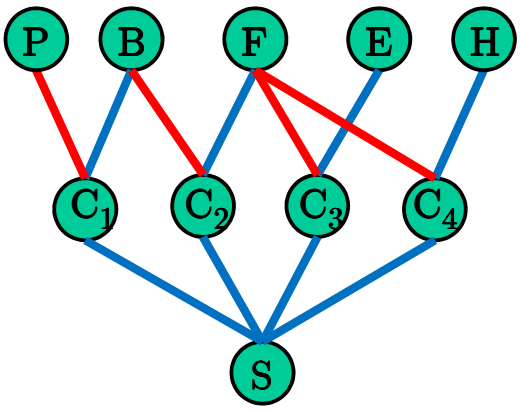
$P(S=1) = \frac{4}{33}$

$n = 2^5 \frac{4}{33} = \frac{32}{33} 4 \approx 4$

Counting: How many assignments satisfy this 2-SAT example
Approximate answer: Use the Monte Carlo Method.

i. **Sample P, B, F, E and H by flipping a coin for each variable N times.**
ii. **For each sample of (PBFEH), check for satisfiability.**
iii. **The probability of satisfiability, P(S=1), is approximated as the number of satisfying samples divided by N.**
iv. **The expected number number of satisfiable samples n = P(S=1) 2^5.**

From max-2-SAT to Energy

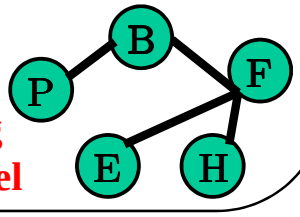


Assume some clauses are harder to satisfy than others. Introduce θ to weigh importance of clauses.

$\neg P \vee B \longrightarrow \theta_1 P(1-B)$
$\neg B \vee F \longrightarrow \theta_2 B(1-F)$
$\neg F \vee E \longrightarrow \theta_3 F(1-E)$
$\neg F \vee H \longrightarrow \theta_4 F(1-H)$

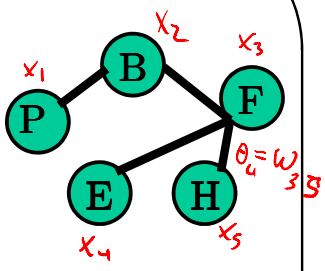
$E = \theta_1 P + \theta_2 B + (\theta_3 + \theta_4) F - \theta_1 PB - \theta_2 BF - \theta_3 FE - \theta_4 FH$

Ising model



From max-2-SAT to Energy

$E = \theta_1 P + \theta_2 B + (\theta_3 + \theta_4) F - \theta_1 PB - \theta_2 BF - \theta_3 FE - \theta_4 FH$
Let $P = x_1, B = x_2, F = x_3, E = x_4$ and $H = x_5$

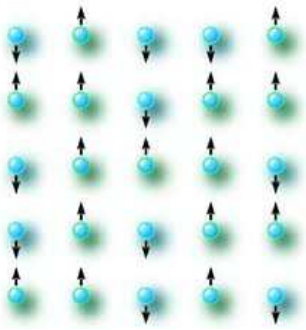


The energy can be written as:

$$E = -\sum_{i=1}^{5} b_i x_i - \sum_{i=1}^{5} \sum_{j>i} x_i w_{ij} x_j$$

In our case:

$b_1 = -\theta_1$ | $w_{12} = \theta_1$
$b_2 = -\theta_2$ | $w_{23} = \theta_2$
$b_3 = -\theta_3 - \theta_4$ | $w_{34} = \theta_3$
$b_4 = 0$ | $w_{35} =$
$b_5 = 0$ | $w_{45} =$



From max-2-SAT to Energy to Probability

Let us look at the energy of a few configurations, assuming all the $\theta_i = 1$.
In this case the energy is simply:

$$E(x_1, x_2, \ldots, x_5) = x_1 + x_2 + 2x_3 - x_1x_2 - x_2x_3 - x_3x_4 - x_3x_5$$

What is the lowest energy? When is it attained?
What is the maximum energy?
What should the most probable configuration be?

x_1	x_2	x_3	x_4	x_5	E
1	1	1	1	1	0
0	1	1	1	1	0
1	0	1	1	1	1
1	1	1	0	0	2
0	0	1	1	1	0

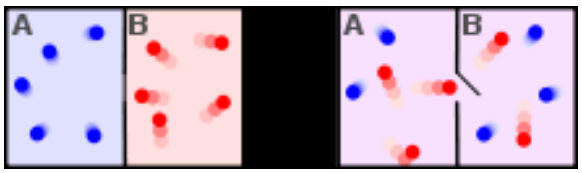
$P(x_1 x_2 x_3 x_4 x_5) = \frac{e^{-E(x_1 \cdots x_5)}}{Z}$

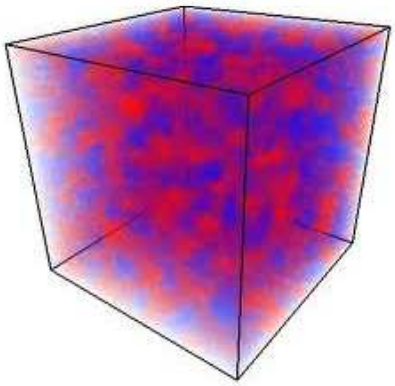
$1 = E(1\,0111) < E(1110\,0) = 2$

$Z = \sum_{x_1} \sum_{x_2} \cdots \sum_{x_5} e^{-E(x_1, \ldots x_5)}$

Ising models and the 2nd law of thermodynamics

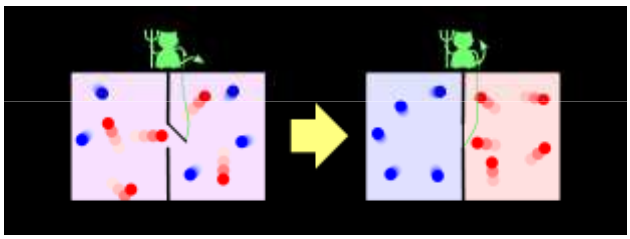
- **The Ising model describes many physical phenomena:**
  - "*The Ising model can be reinterpreted as a statistical model for the motion of atoms. A coarse model is to make space-time a lattice and imagine that each position either contains an atom or it doesn't.*" **Wikipedia Ising Model page.**
  - "*The original motivation for the model was the phenomenon of magnetism.*"
- **Second law of thermodynamics and stability.**





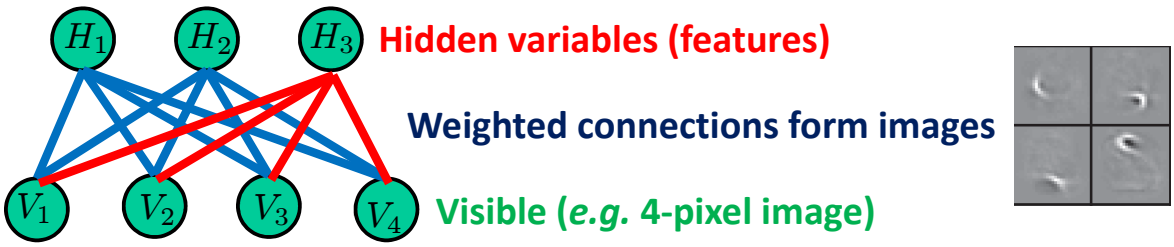
On information and energy – Maxwell's Demon

In this thought experiment, "*an imaginary container is divided into two parts by an insulated wall, with a door that can be opened and closed by what came to be called "Maxwell's Demon". The hypothetical demon is only able to let the "hot" molecules of gas flow through to a favored side of the chamber, causing that side to appear to spontaneously heat up while the other side cools down.*"



- **Does this violate the 2nd law?**
- **What is the relation of information and energy?**

Restricted Boltzmann Machines



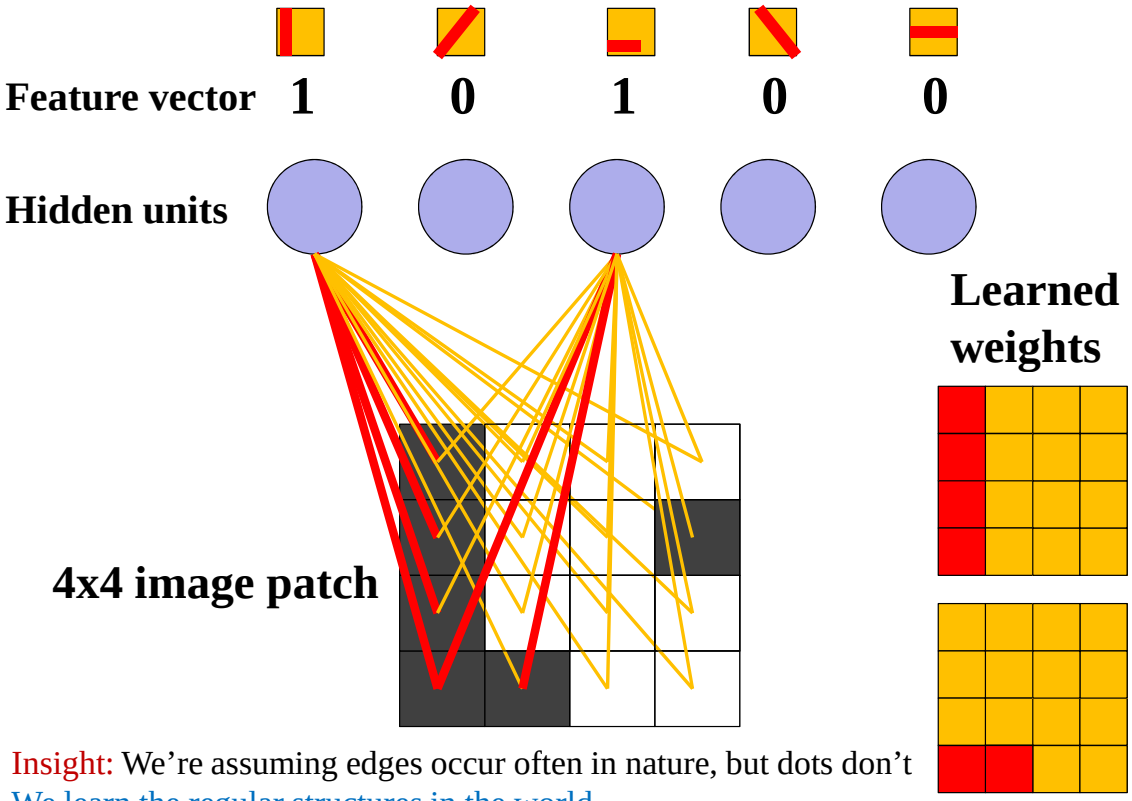
A joint configuration $(\mathbf{v}, \mathbf{h})$ of the binary visible and hidden units has an energy given by the following RBM model:

$$E(\mathbf{v}, \mathbf{h}) = -\sum_{i \in pixels} b_i v_i - \sum_{j \in features} b_j h_j - \sum_{i,j} v_i w_{ij} h_j$$

And hence a Boltzmann probability:

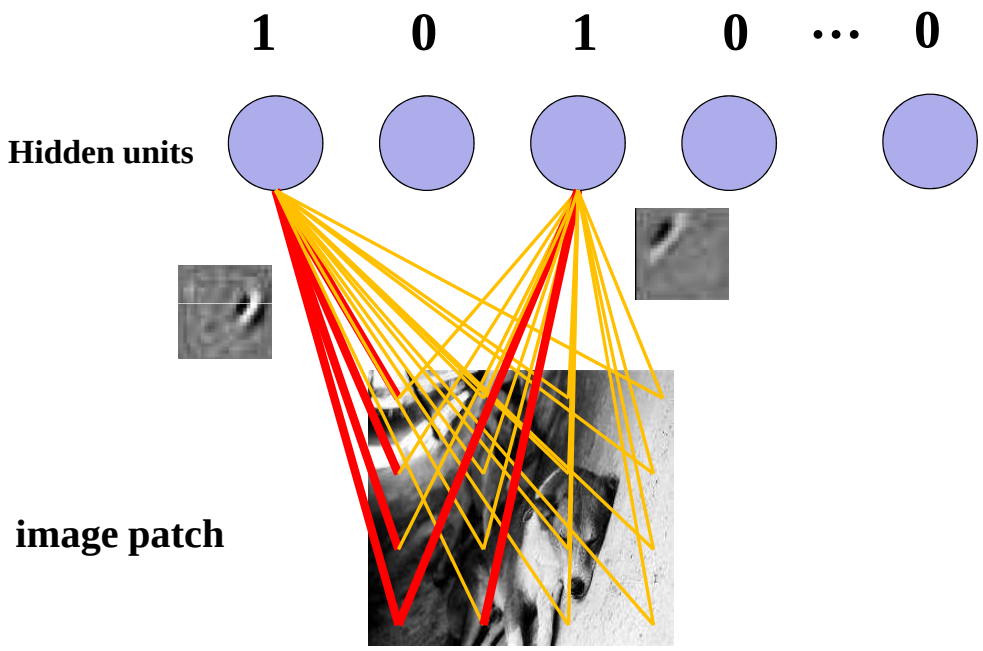
$$p(\mathbf{v}, \mathbf{h}) = \frac{1}{Z} e^{-E(\mathbf{v}, \mathbf{h})}$$

Distributed representation

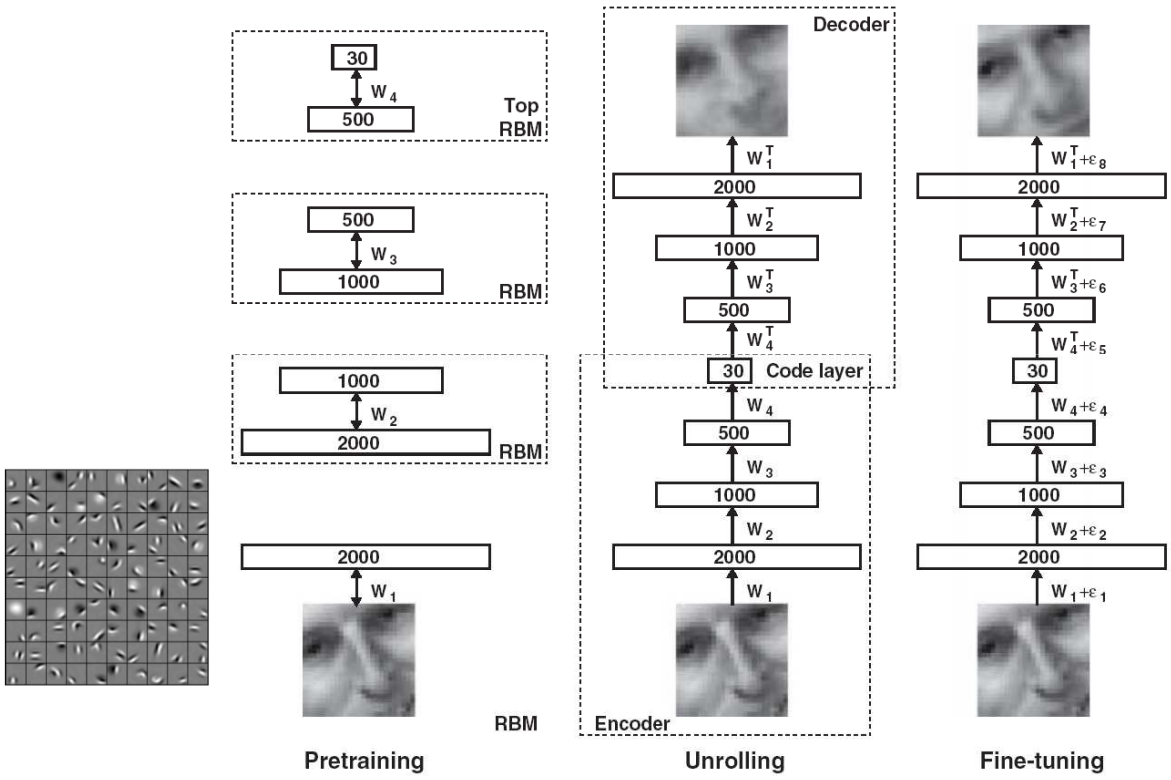


Insight: We're assuming edges occur often in nature, but dots don't
We learn the regular structures in the world

Semantic Hash

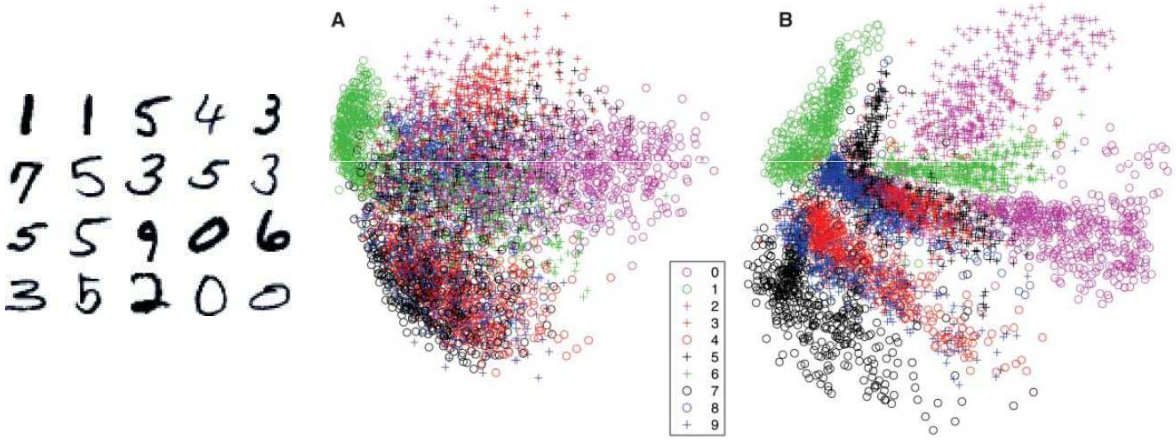


Deep learning (Hinton and collaborators)



Encoding digits

(A) The two-dimensional codes for 500 digits of each class produced by taking the first two principal components of all 60,000 training images.
(B) The two-dimensional codes found by a 784-1000-500-250-2 autoencoder.



These 2-dimensional embeddings of images of digits enable us to make predictions (classification)

In the binary case where $v \in \{0,1\}^D$ and $h \in \{0,1\}^K$ the energy function can be expressed as:

$$E(v,h,W) = -\sum_{i=1}^{D}\sum_{j=1}^{K} v_i W_{ij} h_j - \sum_{i=1}^{D} v_i b_i - \sum_{j=1}^{K} h_j b_j.$$

The probabilities of each node can be easily obtained.

$$p(v_i = 1|h,W) = sigmoid\left(\sum_{j=1}^{K} W_{ij} h_j + b_i\right)$$

$$p(h_j = 1|v,W) = sigmoid\left(\sum_{i=1}^{D} W_{ij} v_i + b_j\right),$$

where $sigmoid(a) = \frac{1}{1+exp(-a)}$. The model is therefore easy to sample: One simply flips K coins for the hidden units and D coins for the visible units.

Contrastive divergence learning

1. Sample hidden units $\widetilde{h_n}$ from $p(h|v_n, W^{(t)})$.
2. Sample imaginary data $\widetilde{v_n}$ from $p(v|\widetilde{h_n}, W^{(t)})$.
3. Sample hidden units again $\widetilde{\widetilde{h_n}}$ from $p(h|\widetilde{v_n}, W^{(t)})$.
4. Update the parameters:

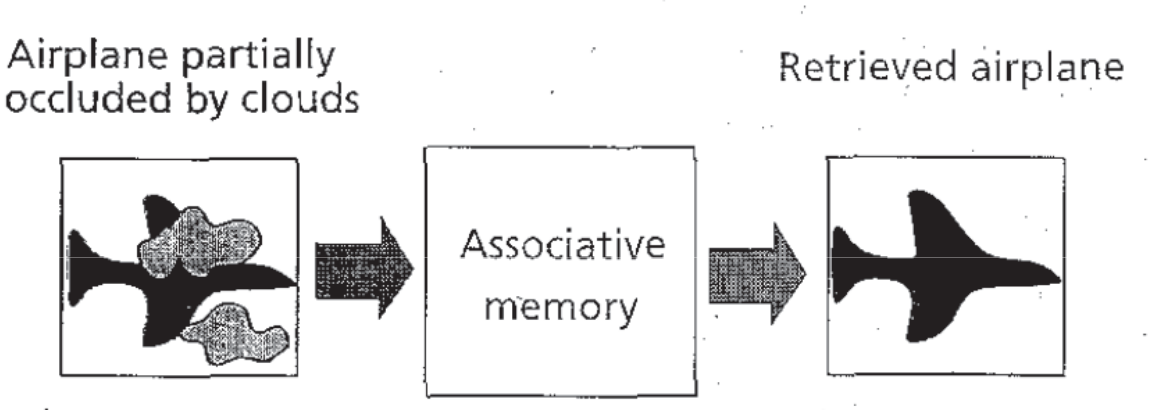
Real data

Confabulation

$$W_{dk}^{(t+1)} = W_{dk}^{(t)} + \eta^{(t)}\left[\frac{1}{N}\sum_{n=1}^{N} v_{dn}\widetilde{h_{kn}} - \frac{1}{N}\sum_{n=1}^{N} \widetilde{v_{dn}}\widetilde{\widetilde{h_{kn}}}\right]$$

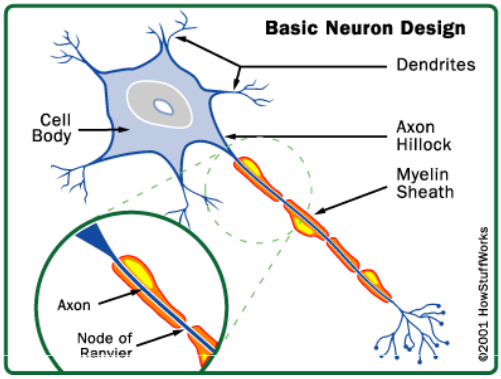
5. Increase t to $t+1$ and go to step 2.

Associative memory



Example 2: Say the alphabet, …. backward

Hopfield models



$$\text{Input to } i = H_i = \sum_{j \neq i} T_{ij} V_j + I_i.$$

$$\text{output} \quad V_i \rightarrow V_i^0 \text{ if } \sum_{j \neq i} T_{ij} V_j + I_i < U_i$$

$$\rightarrow V_i^1 \text{ if } \sum_{j \neq i} T_{ij} V_j + I_i > U_i.$$

Hopfield models

$$E = -\frac{1}{2} \sum_{i \neq j} \sum T_{ij} V_i V_j - \sum_i I_i V_i + \sum_i U_i V_i.$$

The change ΔE in E due to changing the state of neuron i by ΔV_i is

$$\Delta E = -\left[\sum_{j \neq i} T_{ij} V_j + I_i - U_i \right] \Delta V_i.$$

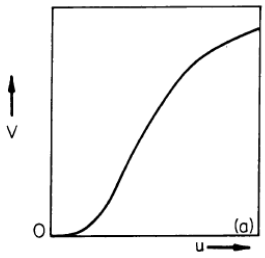
But according to the algorithm, ΔV_i is positive only when the bracket is positive, and similarly for the negative case. Thus any change in E under the algorithm is negative. E is bounded, so the iteration of the algorithm must lead to stable states that do not further change with time.

Hopfield models (systems of ODEs)

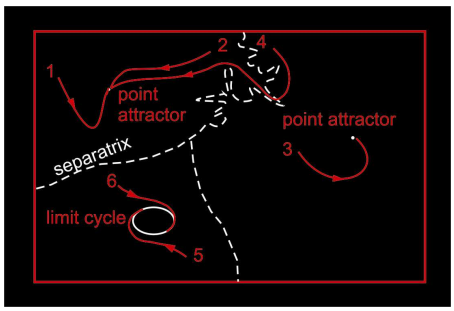
In a biological system, u_i will lag behind the instantaneous outputs V_j of the other cells because of the input capacitance C of the cell membranes, the transmembrane resistance R, and the finite impedance T_{ij}^{-1} between the output V_j and the cell body of cell i. Thus there is a resistance–capacitance (RC) charging equation that determines the rate of change of u_i.

$$C_i(du_i/dt) = \sum_j T_{ij}V_j - u_i/R_i + I_i$$

$$u_i = g_i^{-1}(V_i).$$



Hopfield models (systems of ODEs)



$$E = -\frac{1}{2}\sum_{i,j}\sum T_{ij}V_iV_j$$

$$+ \sum_i (1/R_i)\int_0^{V_i} g_i^{-1}(V)dV + \sum_i I_iV_i.$$

$$dE/dt \leq 0,\ dE/dt = 0 \rightarrow dV_i/dt = 0 \text{ for all } i.$$

time evolution of the system is a motion in state space that seeks out minima in E and comes to a stop at such points. E is a Liapunov function for the system.

Quantum computing (D-Wave)